CAAP Quarterly Report

Date of Report: *July 6, 2019*

Prepared for: *Thomas Finch (Project Manager) and Joshua Arnold (CAAP Program Manager), U.S. DOT Pipeline and Hazardous Materials Safety Administration*

Contract Number: *693JK31850005CAAP*

Project Title: *Low-variance Deep Graph Learning for Predictive Pipeline Assessment with Interacting Threats*

Prepared by: *Hao Zhang (Colorado School of Mines) and Yiming Deng (Michigan State University)*

Contact Information: *Dr. Hao Zhang, Department of Computer Science, Colorado School of Mines; 1500 Illinois St., Golden, CO 80401; Phone: 303-273-3581; Email: hzhang@mines.edu*

For quarterly period ending: *July 7, 2019*

Business and Activity Section

(a) Contract Activity

No contract modification was made or proposed in this quarterly period. No materials were purchased during this quarterly period.

(b) Status Update of Past Quarter Activities

In this reporting period, the research team performed literature review, and made great progress toward achieving the technical objectives including:

(1) Developing heterogeneous data-driven risk identification models (Task 1.2).
(2) Designing methods for spatiotemporal matching of interacting threats (Task 2.2)

In this reporting period, the research team made progress on educational activities, including involving three PhD students and several unpaid master and undergraduate students at Mines and MSU, and adapting the research topics from this project with undergraduate research programs (e.g., the Mines Undergraduate Research Honor Thesis) and MSU (e.g., ENSURE program).

(c) Cost Share Activity

PI Zhang used his 11.29% yearly effort as the in-kind cost share to work on the project at the Colorado School of Mines. Co-PI Yiming Deng used his 6.07% yearly effort as the in-kind cost

share to work on the project at the Michigan State University. The cost share was used following the approved proposal and no modification was made.

(d) Performed Research: Developing and Evaluating New Methods for Low-Variance Interacting Threats Assessment

1. Progress on Task 1.2: heterogeneous data-driven risk identification models

1.1 Mixture Regression & Transfer learning

Multi variate function estimation is effective to determine the growth of the defects when the defects are not interacting with each other. How-ever for setting up a regression model we can consider the response y from grid points containing the locations as grid points.

$$y_l = f(x_{1i}, x_{2j}) \; where \; , \; i, \; j = 1, \cdots, N \; and \; l = 21(i-1) + j \qquad (1)$$

The function f is modelled as a mixture of regression where$K$ represent the number of mixtures:

$$f(x_{1i}, x_{2j}) = \sum_{k=1}^{K} P_k . \quad N(\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2j}, \sigma_k^2) \qquad (2)$$

For each mixture model, the intensity is modeled by a linear surface ($\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2j}, \sigma_k^2$) with aberrations having variability σ_k^2. If σ_k^2 is low then there is not much variability, which will be the case when there are no defects in the vicinity. For areas near defects, the average intensity ($\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2j}, \sigma_k^2$) will be different and leading to the points in those areas being from other mixing densities[4]. Also, for point adjoining defects the variance σ_k^2 will be large. The mixing weight P_k can be made to depend on(x_{1i}, x_{2j}) . While computing, for each point (x_{1i}, x_{2j}) we find out the posterior probabilities $\hat{\pi}_k \; (x_{1i}, x_{2j})$ and based on their value can classify each point to either defective or non-defective areas. If the defective area is not very large, then it is not a harmful defect.
For simplicity, we are using K=2. So, there are only two segments. Now the flux in each segment is modelled by a Normal Regression Model.

- Intensity is modelled by a linear surface $\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2i}$ having variabilityσ_k^2.
- σ_k^2 being low means no defect in the vicinity: as there are spikes in the flux in defective or defect adjoining areas.
- For areas near defects, the average intensity $\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2i}$ will be different and leading to point in those areas being from other mixing densities.
- If two grid points are close the difference in average flux for them in the model is not huge. Thus, spatial relation among points are leveraged.

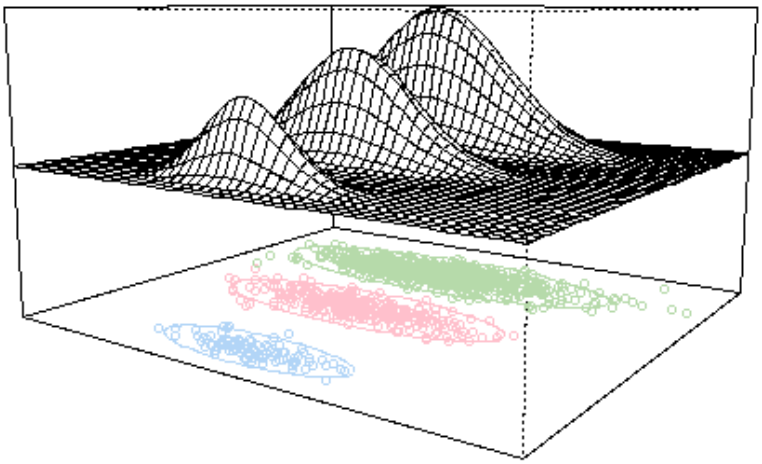


Figure 1: Mixture Regression Model

The objective is to estimate the mixing weights P_k as well as the regression coefficients $\beta_0^{(k)}$, $\beta_1^{(k)}$, $\beta_2^{(k)}$ and regression error σ_k^2. Challenges include no closed form solutions exist. We do not know the probability that the grid point (x_{1i}, x_{2j}) corresponding to l=21(i-1)*j belongs to each of the k regression model.

- **If we knew each of their probabilities** then to estimate the regression coefficients we would be just doing ordinary regression for each of the k models separately abet *now with weights*; The weights signify the probability/amount of membership of each point in each model.
- Denote the weights by $\boldsymbol{p_{lk}}$**:** Prob(the l th point belongs to the k th normal model); l=1,..,21^2; k=1,…,K
- Consider a weight matrix $\boldsymbol{W_k}$ of 441*441 dimensions with $\{\boldsymbol{p_{lk}}$: l=1,…,441} on the di-agonals.
- As the weights are unknown, we use an iterative algorithm. It is called **EM algorithm**.

Expectation Maximization Algorithm and it's details:

- EM is an iterative method to find the maximum likelihood estimates in mixture modeling. It increases the likelihood in each step and there is a lot of literature on the convergence properties of these algorithms[5]
- It comprises of two steps. First consider the M-step.
- In the **M-step** the parameters β, σ are updated for a particular fixed weight matrix $W_j^{(t)}$.
- Once W is fixed the parameters for each model are optimized by **weighted least squares**
- $\beta_j^{(t+1)} = (X^T W_j^{(t)} X)^{-1} X^T W_j^{(t)} y$
- $\sigma_j^{2(t+1)} = \frac{\left\| W_j^{1/2(t)} (y - X^T \beta_j^{(t+1)}) \right\|^2}{tr(W_j^{(t)})}$
- where $\|A\|^2 = A^T A$ and $tr(A)$ means the trace of a matrix A, X is a $441 * 3$ matrix, β_j is a vector $3 * 1$ and response variable y is a vector of length 441.

- $W_j^{(t)}$ is identity matrix, the above is ordinary least squares.
- In the **E-step** the posterior probabilities i.e the weight values $p_{lk}^{(t)}$ are updated.
- Here the $p_{lk}^{(t)}$ signifies the weight of the l[th] response in the j[th] component.
- E-step: $p_{lk}^{(t)} = \left[1 + \sum_{j' \neq j} \frac{P_{j'}^{(t)} \Phi(y_i | x_i^T \beta_{j'}, \sigma_{j'}^2)}{P_k^{(t)} \Phi(y_i | x_i^T \beta_k, \sigma_k^2)}\right]^{-1}$
- This involves finding normal pdf centered at the predicted value $x_i^T \beta_{j'}$ and having variance $\sigma_{j'}^2$.
- Each of them was found was in the **M-step**
- We update likelihood, then the weights and then the mixing weights.

We use these data as representative MFL data pertaining to day 1. Figure 3 shows such data. We represent these data in 2D-images for subsequent processing and analysis (see Fig. 2). Then, for successive days we randomly increase the size of the defective area and obtain flux reading corresponding to it. To produce representative noisy data for these days, we contaminate these MFL readings from COMSOL with Gaussian noise of strength 70% at random places. Figure 4 shows representative data for 6 days where the defect has grown from 6mm to 1cm from day1 to day 6. Also, Day 1 data is much less noisy then data from the subsequent days.

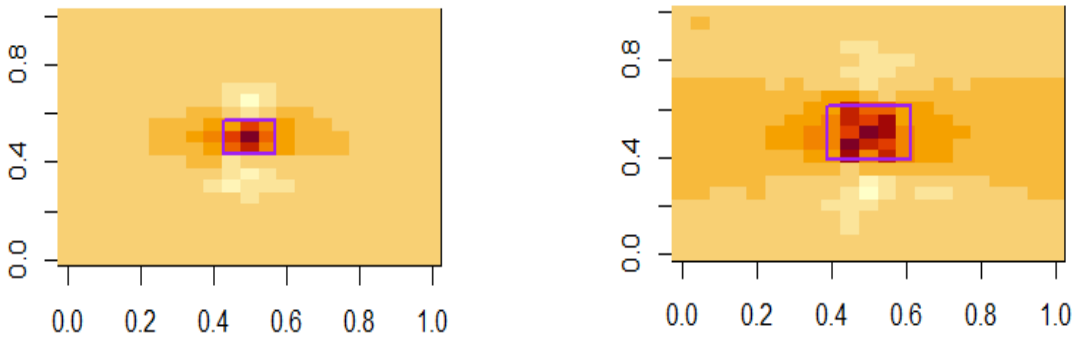


Figure 2 : 2-D Plot of MFL data as locations vary in x and y axes.

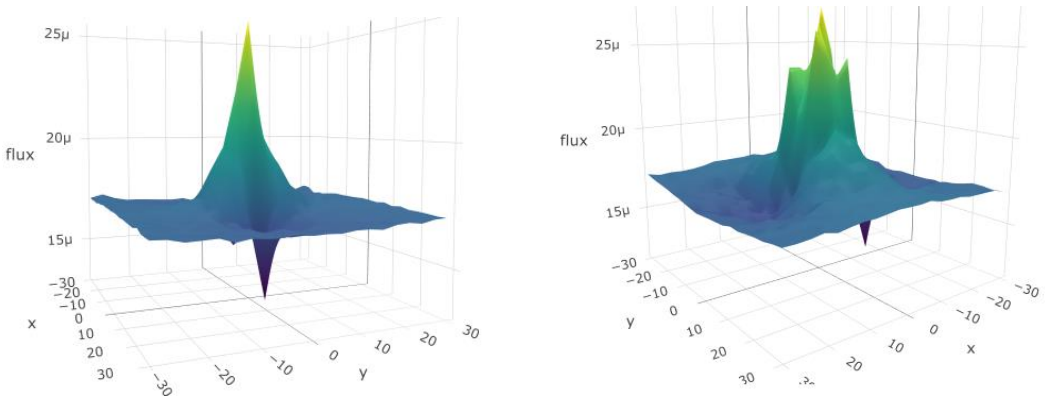


Figure 3: 3D-Plot of MFL data from noiseless Day 1 readings and the bottom one shows a larger defect

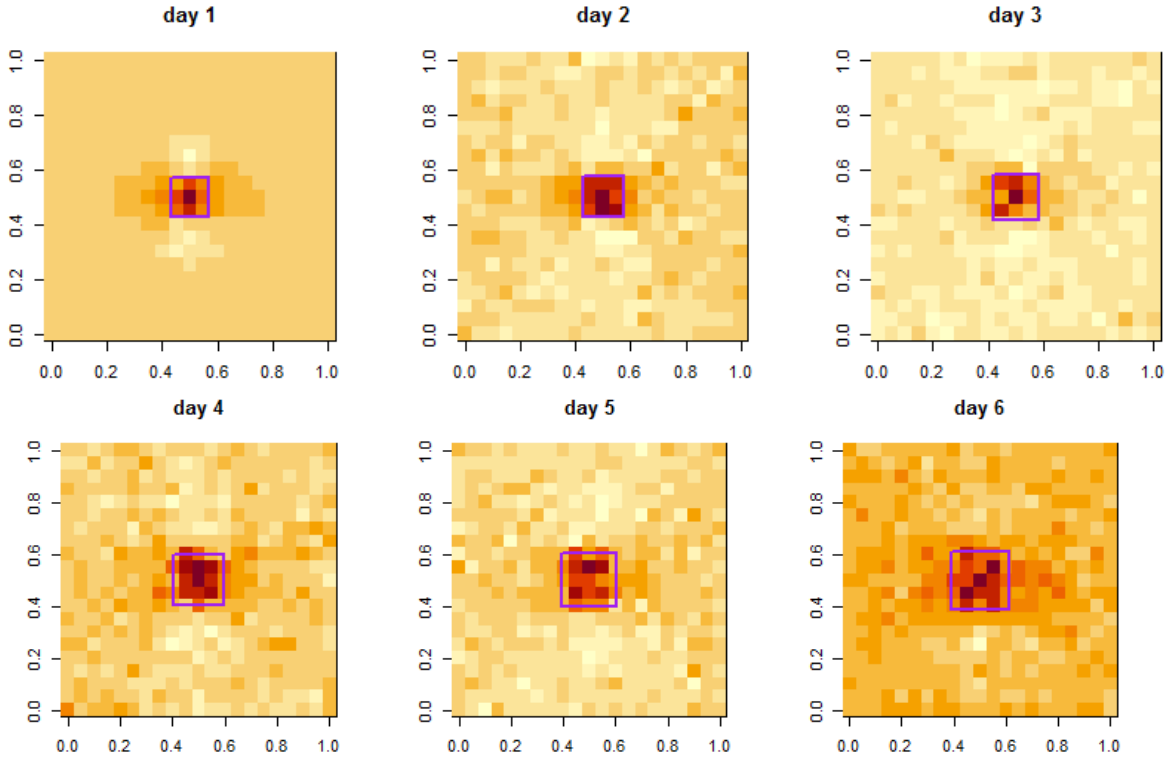
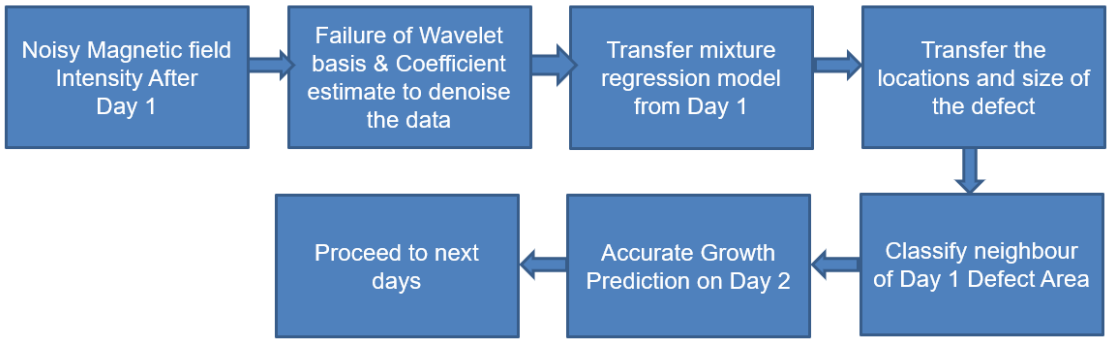


Figure 4: Noisy data on the corresponding days

The steps considered for monitoring the defect growth is as follows:



While computing, for each point (x_{1i}, x_{2j}) we find out the posterior probabilities of defects $\hat{\pi}_D\ (x_{1i}, x_{2j})$ and based on their value can classify each point to either defective (D) or non-defective (S) areas. We considered different kinds of defects such as scratches, very small, moderate and large defect sizes. In all these cases we have observed that our function estimation matches well with the intensity plots and the defective areas can be recognized when we are applying the function estimation on the data provided by Day 1 when it is un-affected by noise. We use the R package mix tools for fitting these w-mixture regression functions. Figure 5 shows the working on a representative day 1 data where the defect is a cube of length 6mm in a 6 cm pipe sector.

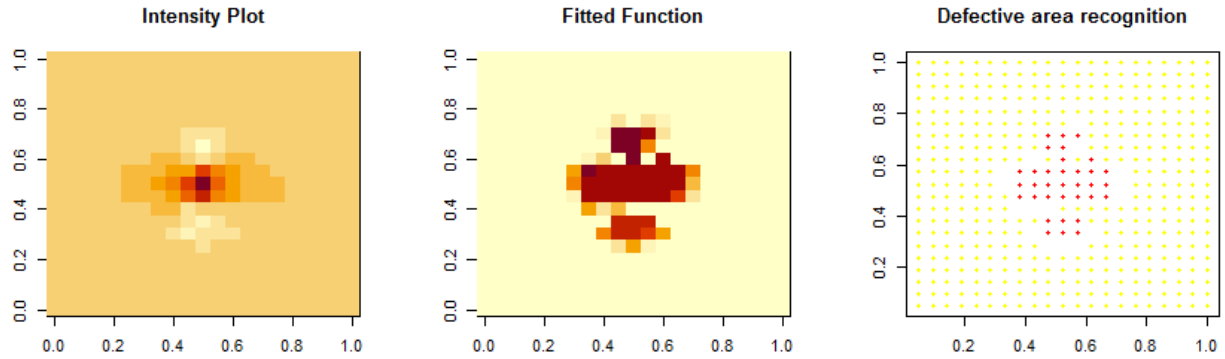


Figure 5: Defect area recognition based on accurate Day 1 readings

Figure 6 shows the defective area identification based on accurate day 1 readings. The defect is a cube of 6 mm. On left, we have the MFL data and the fitted function by the two-mixture regression model in the middle. Based on the thresholding the posterior probabilities at 0.8, the estimated defective scan points are plotted in red.

Now the Day 2 data has been infected with noise. Here mixture regression fails to predict the defect due to presence of high noise. In the below figure the left most figure represent the noisy data from Day 2, the next one is the Day 2 data after denoising. The next plot shows the predicted values based on the two-mixture regression model. Finally, on the extreme right we have in red the detected defect points demarcated in red. The 2-mixture regression model has completely failed to detect the true defect area.

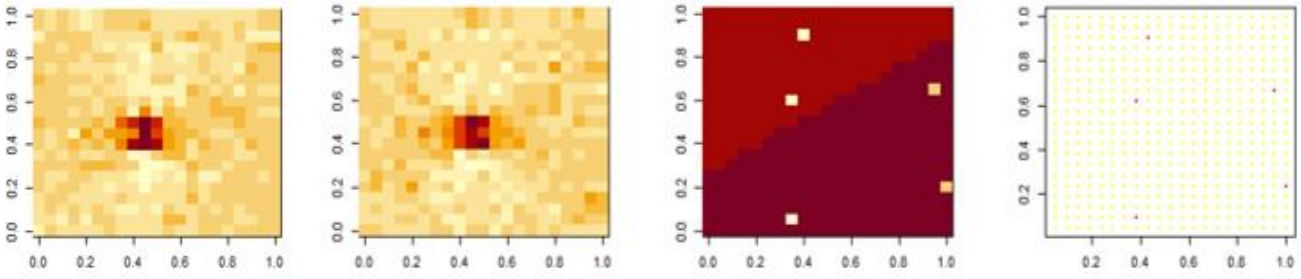


Figure 6: Mixture regression fails due to high noise

Transfer Learning:

On Day 1 points with posterior probabilities of defect greater than 0.8 are considered in the defect area. In Day 2 we mark the neighborhood of defective points. The defect can spread here. But, the scan points outside are safe and are marked as non-defect area. In the below figure based on the posterior probabilities obtained from mixture regression of Day 1 the white space within the yellow ones is the defect and the green ones the neighboring points of the defect [6].

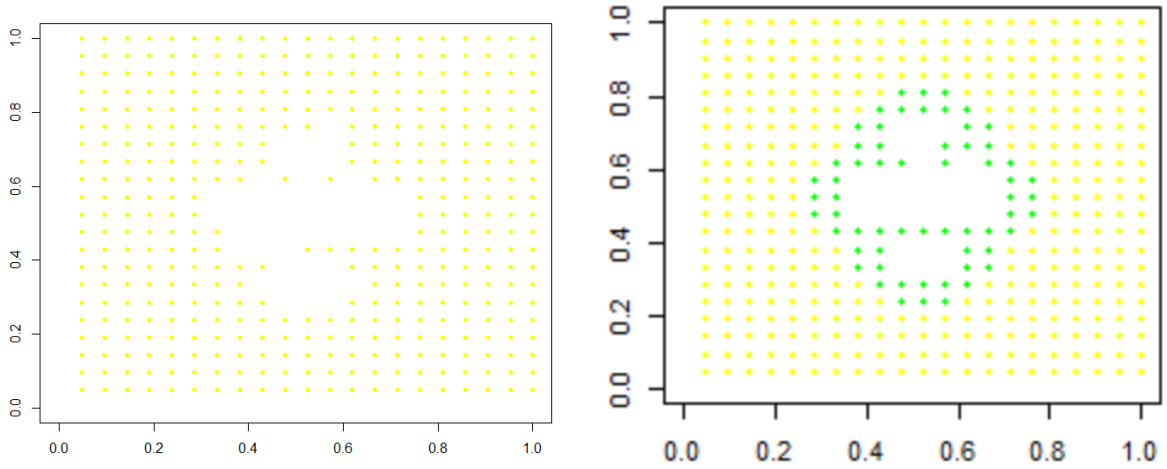


Figure 7: Defect area recognition and neighborhood of the defective area based on Day 1

As Day 2 data is noisy unlike Day 1, hence we do the following: Project Day 1 locations to Day 2 Noisy MFL and flag its neighborhood.

Regress:

- Day 2 MFL values in red defect area by Day 1 defect area MFL
- Day 2 MFL values in yellow area by Day 1 non-defect area MFL

And replace

- Day 2 MFL values in red area by its predicted value from regression.
- Day 2 MFL values in yellow area by its predicted value from regression.

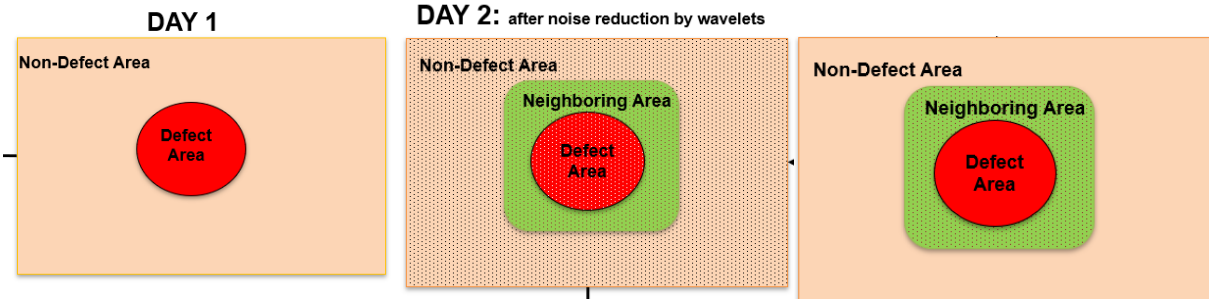


Figure 8: Schematic of projection of Day1 location on Day2

This will make the data smooth in those parts by reducing noise. Now on the transformed data on Day 2 we know that every scan point in the neighboring data has MFL values of Day 2. They are not changed. We also do not know whether these points are defect or not. We fit a 2-mixture regression model on transformed data. For scan points in the neighboring area, we call it defect if its posterior probability of belonging to the defect model is greater than 0.8. Else we consider this as non-defect.

The algorithm is a s follows:

Let D_1 denote the location of defect scan point at day 1:

$$D_0 = \{(x_i, y_i) : \ \hat{\pi}_D\ (x_{1i}, x_{2j}) \ \geq 0.8\} \qquad (3)$$

Consider $Z_1 = \{I_1(x_{1i}, x_{2j}) : (x_{1i}, x_{2j}) \in D_1\}$ as the set of Day 1 MFL values on the defect scan points. Let $Z_2 = \{\{I_2(x_{1i}, x_{2j}) : (x_{1i}, x_{2j}) \in D_1\}$ be the set of Day 2 MFL values on the same scan points which were judged in the defective area on Day 1. Note, that Z_2 is very noisy. Hence it is important to denoise Z_2. This can be done by fitting a linear model on Z_2 based on Z_1 values.

$$Z_{2k} = \alpha_2 + \beta_2 Z_{1k} + \sigma_2 \xi_k \qquad (4)$$

Where $K = 1,2, \ldots, |D_1|$ and $\alpha_2, \beta_2, \sigma_2$ are intercept, slope and standard deviation parameters and $\{\xi_k : k \geq 1\}$ are iid gaussian noise. We find out the least square coefficients $\widehat{\alpha_2}, \widehat{\beta_2}, \widehat{\sigma_2}$ and consider the set of predicted value $Z_2^P = \{\widehat{\alpha_2} + \widehat{\beta_2} Z_1(x_{1i}, x_{2j}) : (x_{1i}, x_{2j}) \in D_1\}$. Now Z_2^P will be much less nosier than Z_2. Note that in Day 1, Z_2 was well modeled by a linear model and so, we can expect Z_2^P which is the denoised version of Z_2.got by projecting it on Day 1 MFL values, to be also well-modelled by linear model. Consider the neighborhood of the defect points of Day 1. Let N_1 denote the neighborhood of the scan points in D_1, i.e.:

$$N_1 = (x_{1i}, x_{2j}) : min_{(x,y)\varepsilon D_{j-1}} |x_{1i} - x| + |x_{1j} - y| < \delta\} \backslash D_1$$

where δ is the tuning parameter which is taken as 3 here.

In this neighborhood points N_1 the defect can spread on Day 2. This is the uncertain area. All points outside N_1 and D_1 can be considered to be safe on Day 2. Let S_2 denote the set. So, $S_2 = (N_1 \cup D_1)^c$. Let $W_1 = \{I_1(x_{1i}, x_{2j}) : (x_{1i}, x_{2j}) \in S_2\}$ and $W_2 = \{I_2(x_{1i}, x_{2j}) : (x_{1i}, x_{2j}) \in S_2\}$ be the MFL data from Day 1 and 2 respectively on these safe points. Again, as W_2 is noisy so we fit a linear model say W_{2k} and consider the predicted values based on the least square coefficients as W_2^P. Hence by projecting on day 1 data, we have denoised and sort of linearized Z_2^P and W_2^P. So, the MFL value in Day 2 for all scan points other than the neighborhood points are well regularized by this transfer learning approach. Let U_2 denote the set of Day 2 MFL values in the neighborhood points N_1 i.e $U_2 = \{I_2(x_{1i}, x_{2j}) : (x_{1i}, x_{2j}) \in N_1\}$. We now fit a mixture regression model with 2 groups on $\{Z_2^P, W_2^P, U_2\}$ response values. As the proportion of points in $(S_2 \cup D_1)$ is much higher than N_1points, we have much more denoised points in $\{Z_2^P, W_2^P, U_2\}$ than noisy points. Hence, a mixture regression model, can't be fitted accurately before. Using the model which is basically trained by the points in Z_2^P and W_2^P., we look at the posterior of points in N_2 and classify them as safe or defect. So, the D_2 defect area is : $D_2 = D_1 \cup \{((x_{1i}, x_{2j}) \in N_1 : \widehat{\pi_2^D}((x_{1i}, x_{2j}) \geq 0.8\}$.

Similarly, for Day 3, we consider the neighborhood N_2 of the defective area D_2 of day 2 and proceed in similar way as before.

In the below figure 9 the green points denote the defect projection from Day 1 to day 2, the yellow ones represent the non-defect ones from the day 2 whereas the purple ones are the non-determined ones which can either be defect or non-defect.

In the Figure 10 (a), the red points show the portion of the defects whereas in figure 10(b) it shows that most of the purple points have changed to yellow and some to red, thereby reducing the uncertainty.

Figure 12 show if we know the defect size then the arrow marks shows the defects from actual data thereby showing the accuracy.

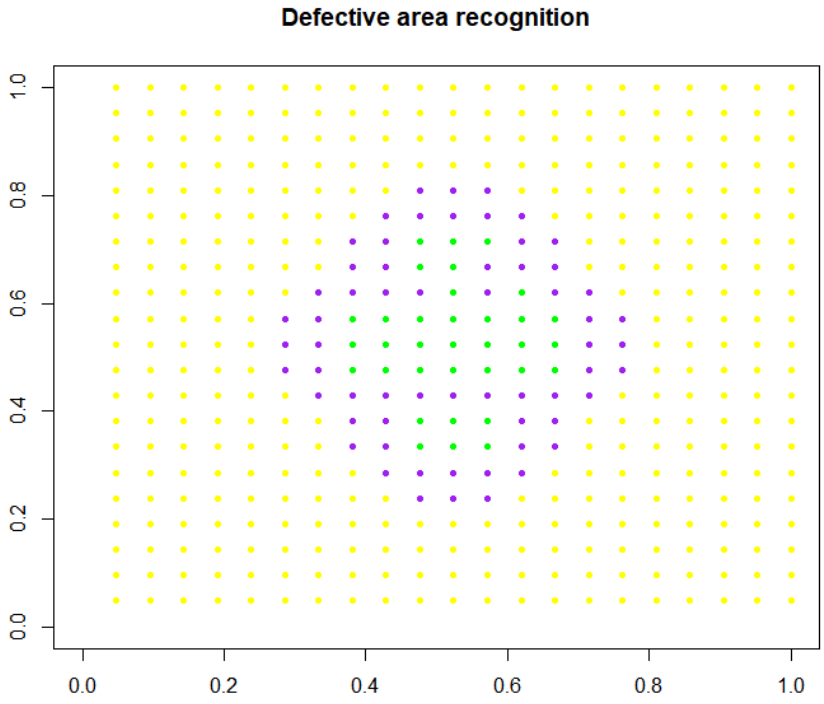


Figure 9:Projection of day 1 on day 2

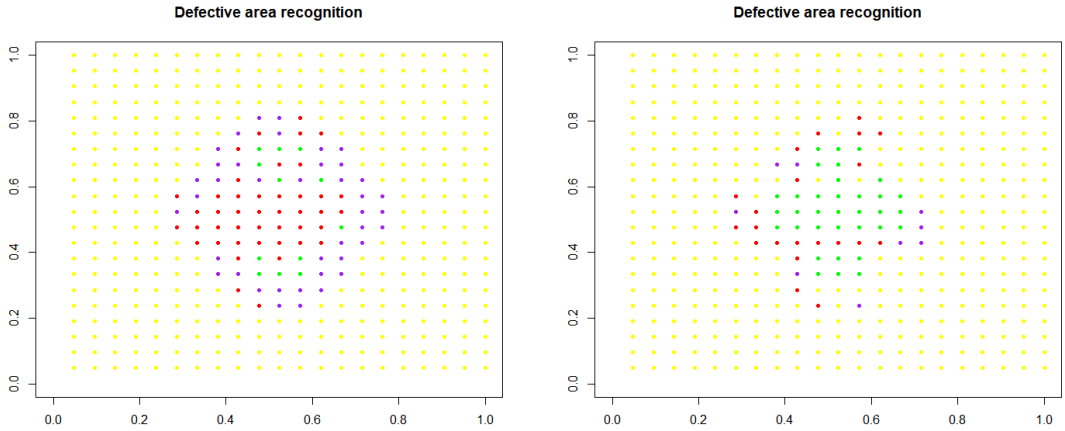


Figure 10: Showing the neighborhood areas with uncertainty reduction

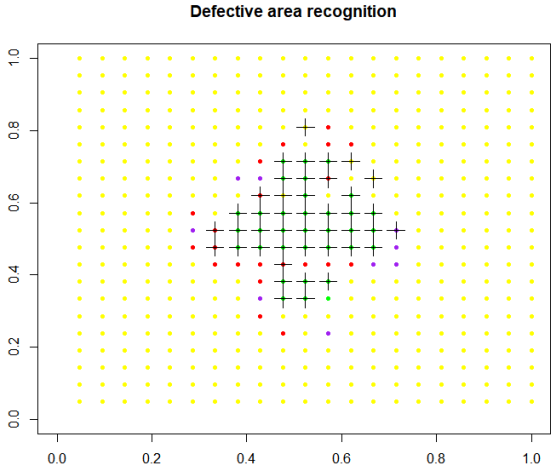


Figure 11: Comparison with actual data

The same process is conducted for the later days too.

However, when the defects are interacting among each other, formerly the mixture regression will not function properly as here the variance σ_k^2 is in the form of identity matrix. But when two or more defects are interacting with each other then σ_k^2 will not be an identity one as there will be co-relation among the defects. Hence for studying the interaction we have to formulate our algorithm as a function of distance and that's where **kriging** comes into play.

- **Regression Kriging [7]:** RK is a spatial prediction technique that combines a regression of the dependent variable on auxiliary variables with kriging of the regression residuals. It is basically the interpolation method called *kriging*, where auxiliary predictors are used directly to solve the kriging weights.
- Kriging or Gaussian process regression is a method where the interpolated values are modelled by a Gaussian process governed by prior covariance.

- The main idea behind kriging is to predict the value of a function at a given point by computing a weighted average of the known values of the function in the neighborhood.
- In our model also, if two grid points are close, the difference in average flux for them in the model is not huge. Thus, spatial relation among points are leveraged and hence due to this spatial contiguity kriging can be applied.
- Kriging is basically a form of Bayesian inference where it starts with prior distribution of functions. The prior takes the form of a Gaussian process, where the samples from the function will be normally distributed and the covariance between any two samples is the covariance function of the Gaussian process evaluated at the spatial location of two points.
- Now, a new value can be predicted at any new spatial location, by combining the Gaussian prior with a Gaussian likelihood function for each of the observed values. The resulting posterior distribution is also Gaussian, with a mean and covariance that can be simply computed from the observed values, their variance, and the kernel matrix derived from the prior.
- The main objective is to produce conditional simulations and compare their variability. The final outcome will be the relevant prediction and uncertainty evaluation.
- Here the two-dimensional coordinates on a grid will be considered.
- Theory of Random Fields: There are some processes that operate over an entire region of interest. These can lead to regional trends, where an attribute can be modeled as a function of the geographic coordinates. However, the regional trend often does not explain all the variation. One must focus on local spatial dependence to determine processes that affect the neighborhood. There is a deterministic local process that causes nearby points to be alike [8, 9].
- The basic difference between the kriging and mixture regression is as follows: In mixture regression the variance is an identity matrix but for kriging after finding the trend of mean, we have to find out σ, the error correlation as if the threats are interacting then the σ will not be an identity matrix.
- Key idea that nearby observations may be correlated is incorporated here.
- Just like any other variables, the set of random variables making up the regionalized variable may have covariance, i.e. one may be related to another (positively or negatively).
- Now each point being a different realization, hence they are different variables hence they must have a covariance.
- Under certain assumptions, this covariance can be considered to depend only on separation between the point's key insight in local spatial dependence.
- Hence Kriging is utilized for mapping of surfaces from limited sample data from the estimation of values at unsampled locations. This is analogous in predicting the interaction of the defects, where we know the defect locations which can be treated as sampled points and predicting their interaction can be treated as unsampled points in spatial domain. There are several build-in packages in R which converts our data to spatial domain.
- A semi-variogram models the difference between a value at one location and the value at another location according to the distance and direction between them.
- In ordinary kriging a weighted average of neighboring samples is used to estimate the 'unknown' value at a given location where the weights are being optimized using a variogram model.

- Say, the realization of a variable Z at spatial locations(s) can be considered as the result of three distinct processes: $Z(s) = m(s) + E`(s) + E''$ where $m$ is the regional trend, $E`$ a spatially correlated stochastic process and E'' is the noise which is neither spatially-correlated nor deterministic.

Steps for implementing Kriging:

- First, we have to load the dataset in the form of data frame.
- Convert the data frame into spatial points data frame.
- Fit a variogram model to the data.
- Perform Interpolation. It is basically the step to estimate values at points we don't have measurements for based on the points for which we do have measurements. Hence, we need two spatial domains, one having values associated with the points, and one for which we want estimates.
- Finally, Kriging is to be performed.

1.2 Modelling of Pulsed eddy current inspection

Conventional eddy current techniques have been used to a great extent for detection of surface breaking defects in conductive materials. However, detection of sub-surface defects is limited due to the single frequency and skin effect phenomena. Pulsed Eddy Current (PEC) techniques excite the probe's driving coil with a repetitive broadband pulse, usually a square wave. The resulting transient current through the coil induces transient eddy currents in the test piece, these pulses consist of a broad frequency spectrum, and the reflected signal contains important depth information. As an alternative technique to conventional eddy current NDT, pulse eddy currents (PEC) is employed which is usually excited by means of a non-sinusoidal coil current.

As with conventional eddy current, a nearby sample and geometrical properties - such as conductivity, magnetic permeability, size and shape - affect measured transient eddy current signals. However, in the Fourier sense, transient signals are far richer in information than are their conventional single-frequency counterparts, since they contain the sample's response to an infinite set of frequencies. Furthermore, transient excitations may be tailored to contain a particular distribution of frequencies with the potential of enhancing sensitivity to specific features, such as material thickness, liftoff distance and material defects. It has been determined that, in contrast to conventional eddy current signals, the presence of ferromagnetic materials may enhance flaw detection rather than impair it. In most systems, a steady state current is allowed to persist for some time before the waveform repeats. The steady state period is made sufficiently long so that all eddy currents have completely decayed away to undetectable levels. This is also important when we integrate PEC with MFL.

Finite Element Modelling (FEM) of PEC

Model definition

For FEM of PEC technique, COMSOL Multiphysics 5.3a [10]has been adopted. In this study, symmetrical probes and defects have been considered. The problem has been assumed to be linear to further reduce the computational complexity. The boundary of internal structure is set to Dirichlet condition while external boundaries of the model are set to Neumann condition on the magnetic field. The problem domain is discretized using triangular elements and the governing equation below is solved by applying the above-mentioned boundary conditions.

$$\nabla \times \left(\frac{1}{\mu}\nabla \times \vec{A}\right) = -\sigma\frac{\partial \vec{A}}{\partial t} - \nabla\sigma V + J_S \qquad (5)$$

where $\vec{A}$ is the magnetic vector potential in Vs/m, V is the electric scalar potential in volts, μ represents the magnetic permeability in H/m, σ is the electric conductivity in S/m and J_S is the source current density in $\vec{A}/m^2$. The problem is considered as linear if μ is assumed to be constant in the range of operating frequencies (<100 kHz) and excitation current (<1 A). The problem is solved using transient time stepping mode Direct linear solver. A typical driver-receiver PEC probe consists of a primary coil, a secondary coil and a ferromagnetic core. Here for simplicity and compatibility, ferromagnetic core is replaced with an air core. The primary coil receives the AC electrical input signal from external circuit. As a result of mutual induction, an induced voltage is obtained across the secondary coil.

The model assumes that the primary and secondary windings are made of thin wire and have multiple turns. Using the assumptions that the wire diameter is less than the skin depth and that there are many turns, these windings are modeled with Coil features. Furthermore, the model does not account for eddy currents in the individual turns of the coil. The primary winding is connected to a resistor R_P and the voltage source while the secondary winding is connected to the secondary load resistor, R_s as shown in Fig.12

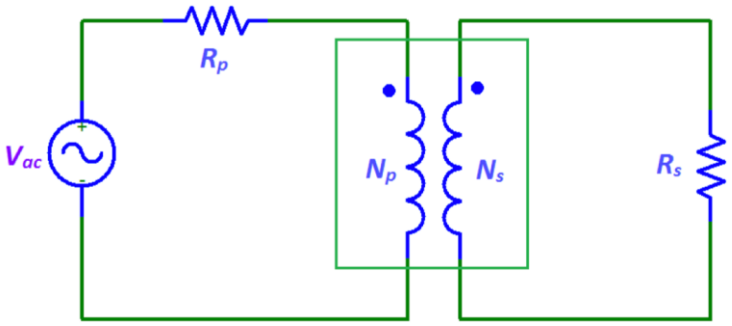


Figure 122: PEC probe with primary coil connected to an external circuit with voltage source and resistor, secondary coil connected to another resistor.

1.3 PEC Model Principle [11]

According to Biot-Savart law, the magnetic flux density of an arbitrary point P produced by a current that flows along line C is integral of the magnetic flux density dB produced by an elementary current Ids.

$$dB = \frac{\mu_0}{4\pi} \cdot \frac{Ids \times \vec{r}}{r^3}$$

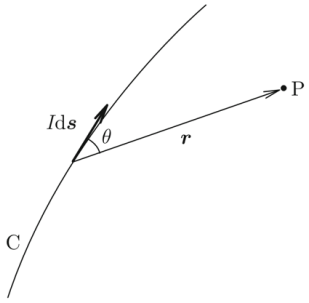
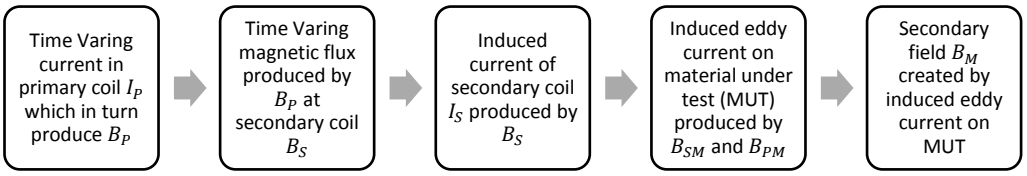


Figure 13: Elementary current along C and observation point P

$$B(\vec{r}) = \frac{\mu_0}{4\pi} \cdot \int \frac{Ids \times \vec{r}}{r^3} \qquad (6)$$

In this way, square circuit with uniform current I, width d will produce the magnetic flux B at a distance z from the circuit surface.

$$B(z) = \frac{\mu_0 I}{4\pi} \cdot \frac{d}{\sqrt{z^2+\frac{d^2}{2}}\sqrt{z^2+\frac{d^2}{4}}} \qquad (7)$$



Time varying current in primary coil will produce a time varying magnetic flux B_P which in turn induce eddy current in the material, and then the induced eddy current will generate the secondary magnetic field B_S. The total magnetic field will form an electric field E in the material, explained by the fundamental Maxwell's Equations:

$$\nabla \times E = -\frac{\partial(B_P+B_S)}{\partial t} \qquad (8)$$
$$J = \sigma E$$

where σ is conductivity of the material. On the other hand, the induced current in the secondary coil is defined as J_S

$$\nabla \times J_S = -\sigma \frac{\partial(B_P+B_S)}{\partial t} \qquad (9)$$

Then the secondary magnetic field can be further defined as follows

$$\nabla \times B_S = \mu_{coil} J_S$$
$$\phi_S = \int B_S \, ds \qquad (10)$$

Model setup and simulation results

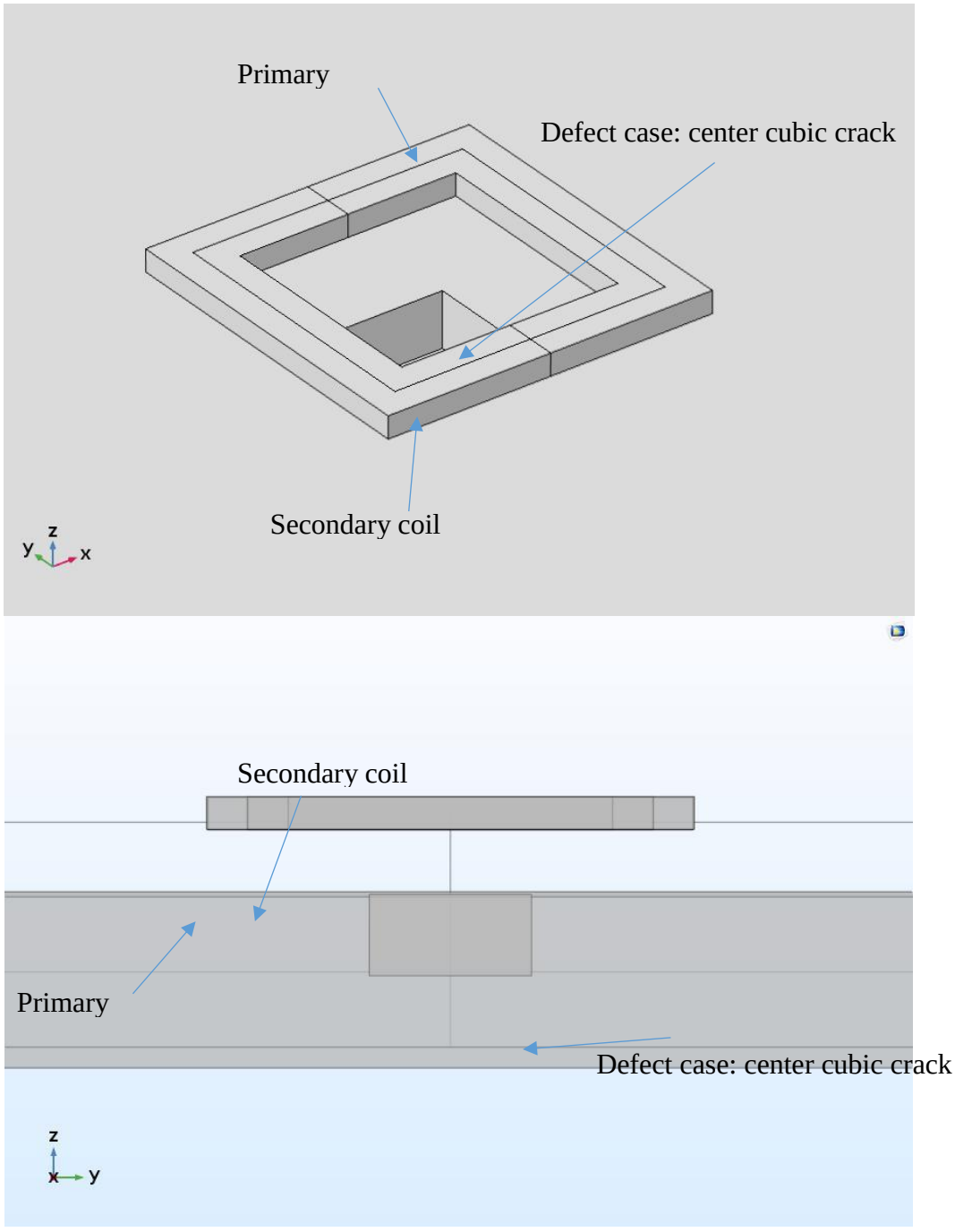


Figure 14: Schematic of PEC probe and MUT with center cubic defect

The 3-D FEM simulations are conducted by COMSOL. The simulation models of PEC coils are shown in Figure above. Here, the winding directions of primary (excitation) coil and secondary coil are clockwise. The electromagnetic parameters and structure parameters of the simulation models are listed in Table below

Quantity	Value
Depth of steel	10 mm
Thickness of PEC coils	2mm
Length of primary coil	20mm
Length of secondary coil	18mm
Width of primary coil	2mm
Width of secondary coil	2mm
Lift-off	3mm
Primary coil turns	100
Secondary coil turns	100

Both non-defect and defect cases are considered. During simulation, size and location of the defect are set as variables. MUT studied here is Q235 steel and the incremental permeability has a nonlinear property, as the B-H and H-B curve shown in Fig. 15.

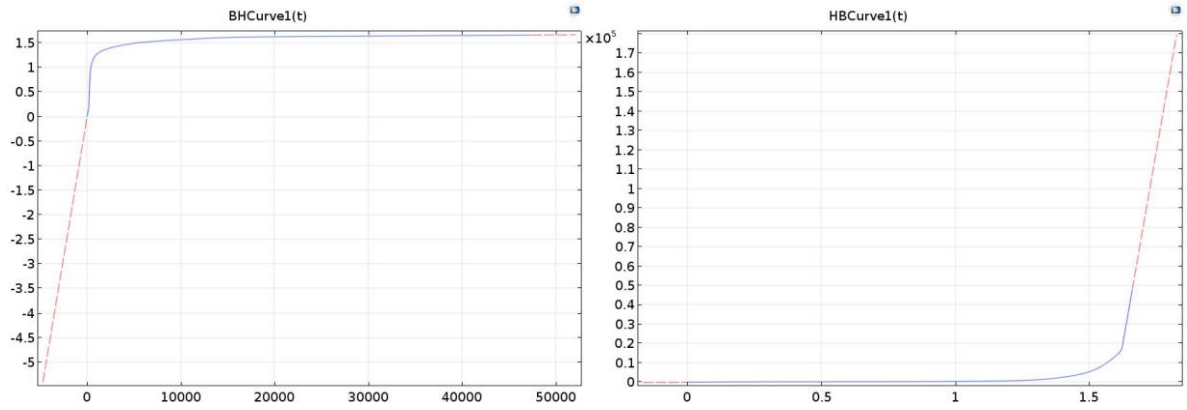


Figure 15: BH and HB curve of the sample

Fig.16 shows the time-domain waveform of the pulse-excited current with direction the same as the winding direction of the primary coil. The excitation time is set from 0 to 10 μ_s , and the pulse width is set to be 2.5 μ_s. Durations of rising edge and falling edge are set to 100 ns.

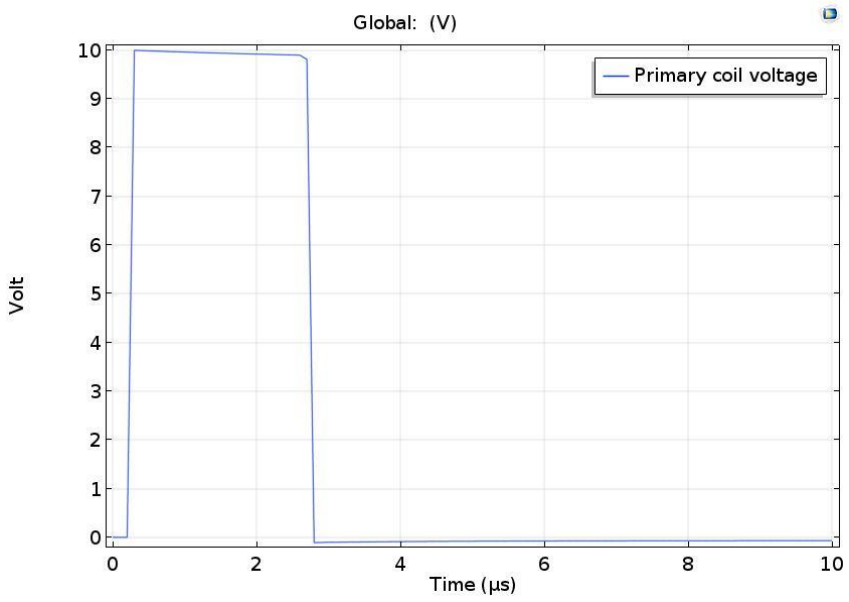
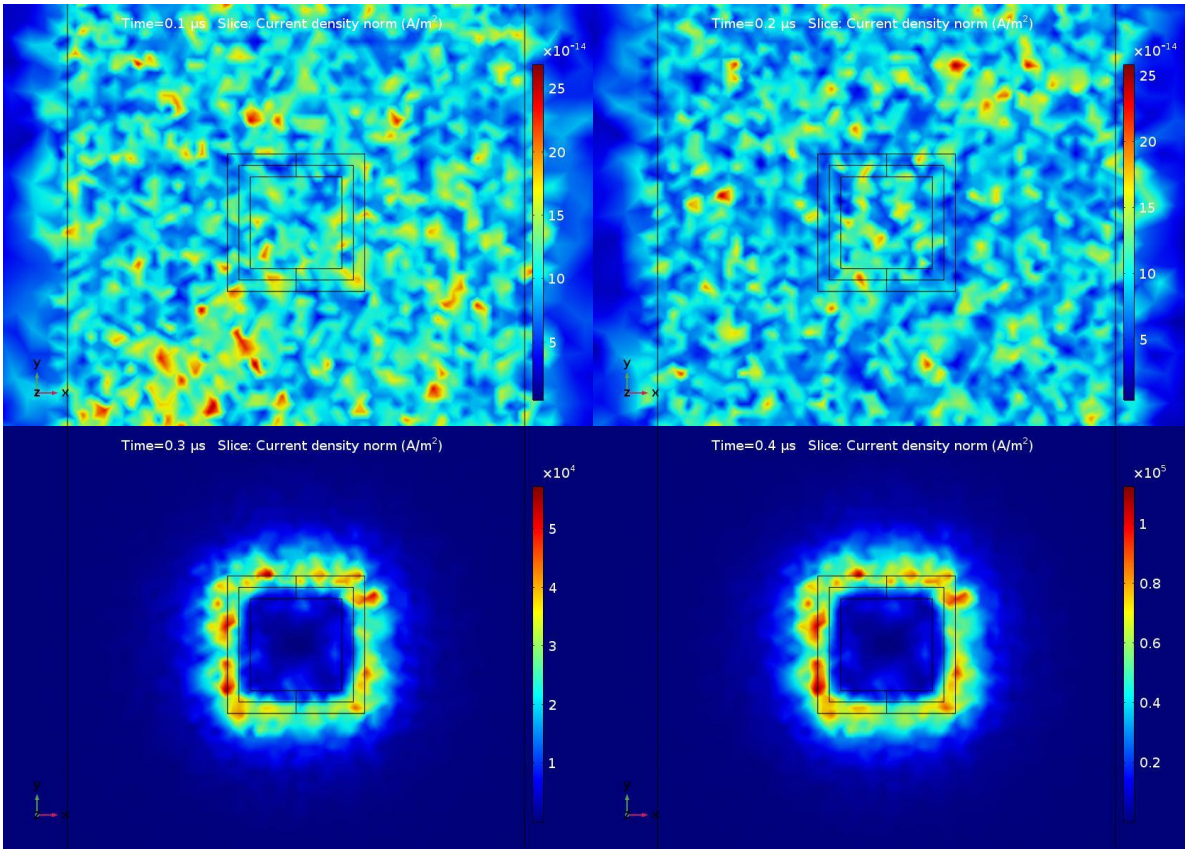
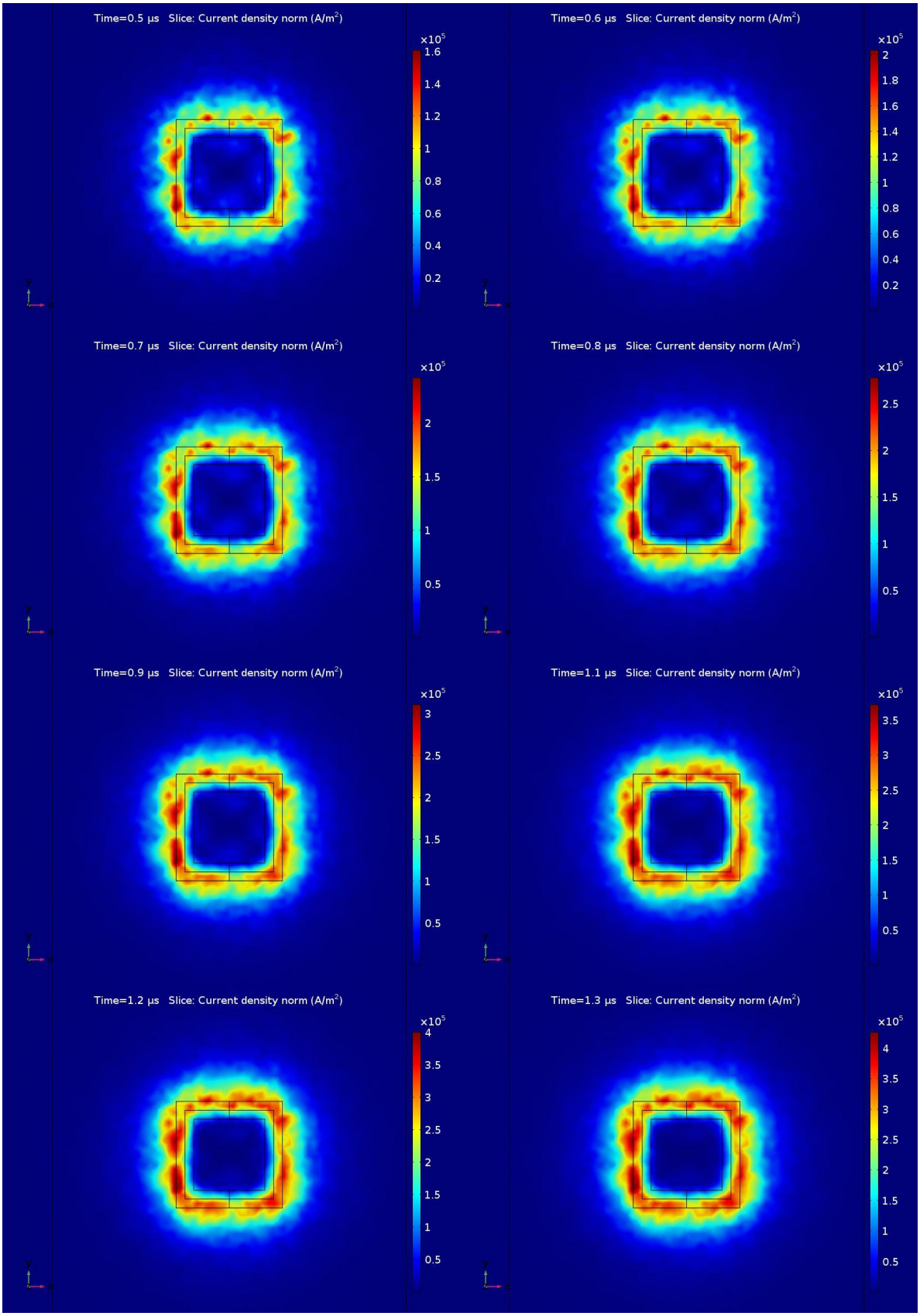


Figure 16: Pulsed signal

For time-dependent study, non-defect case is firstly analyzed. Discretized time-based observation regarding induced current density of MUT upper surface is made during $t = 0$ to $t = 10\ \mu_s$ with step size of 0.1 μ_s at beginning phase.




Time=0.5 μs Slice: Current density norm (A/m²)
×10⁵
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
Time=0.6 μs Slice: Current density norm (A/m²)
×10⁵
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
Time=0.7 μs Slice: Current density norm (A/m²)
×10⁵
2
1.5
1
0.5
Time=0.8 μs Slice: Current density norm (A/m²)
×10⁵
2.5
2
1.5
1
0.5
Time=0.9 μs Slice: Current density norm (A/m²)
×10⁵
3
2.5
2
1.5
1
0.5
Time=1.1 μs Slice: Current density norm (A/m²)
×10⁵
3.5
3
2.5
2
1.5
1
0.5
Time=1.2 μs Slice: Current density norm (A/m²)
×10⁵
4
3.5
3
2.5
2
1.5
1
0.5
Time=1.3 μs Slice: Current density norm (A/m²)
×10⁵
4
3.5
3
2.5
2
1.5
1
0.5

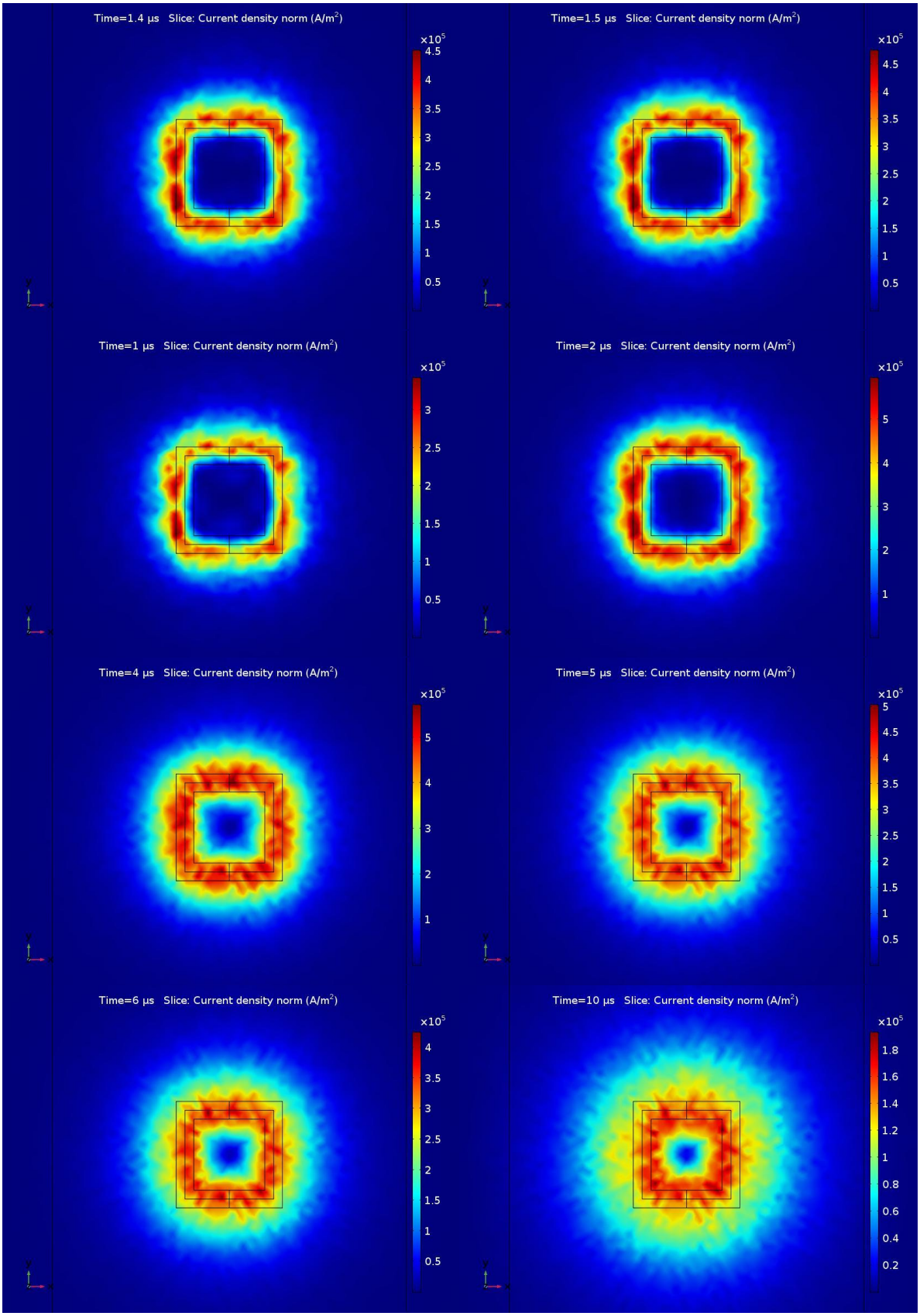

Time=1.4 μs Slice: Current density norm (A/m²)
×10⁵
4.5
4
3.5
3
2.5
2
1.5
1
0.5
Time=1.5 μs Slice: Current density norm (A/m²)
×10⁵
4.5
4
3.5
3
2.5
2
1.5
1
0.5
Time=1 μs Slice: Current density norm (A/m²)
×10⁵
3
2.5
2
1.5
1
0.5
Time=2 μs Slice: Current density norm (A/m²)
×10⁵
5
4
3
2
1
Time=4 μs Slice: Current density norm (A/m²)
×10⁵
5
4
3
2
1
Time=5 μs Slice: Current density norm (A/m²)
×10⁵
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
Time=6 μs Slice: Current density norm (A/m²)
×10⁵
4
3.5
3
2.5
2
1.5
1
0.5
Time=10 μs Slice: Current density norm (A/m²)
×10⁵
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2

Figure 17: Densities and distribution patterns of induced current at upper surface of MUT between $t= 0\mu_s$ and $t= 10\mu_s$

Density of induced current increase rapidly at the beginning due to the magnetic field produced by pulsed signal while decays very slow. Even at $t= 10\mu_s$ when pulsed signal has ceased to 0 for 7 μ_s, current density still remains significant. By placing 3 imaginary sensors (geometry of the sensor won't affect simulation) at the center of MUT with lift-off of 1mm,2mm and 3mm respectively. B_z is measured across three domain points.

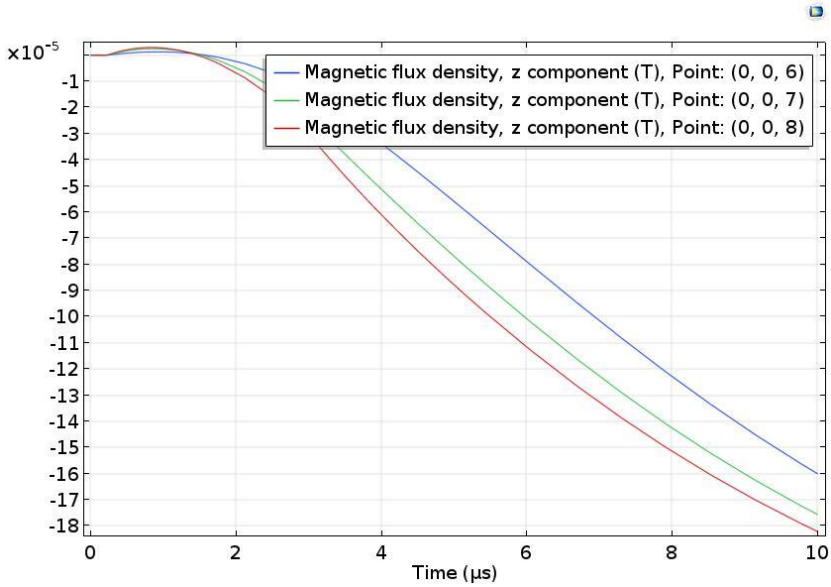


Figure 18: Magnetic flux density with lift-off 1mm,2mm and 3mm

By implementing simulation at a much larger time domain (100 times longer than original inspection), decrease in magnetic flux density along z direction at three locations are observed. As illustrated in figure below, decay rate is much lower which will significantly affect induced signal at secondary coil from MFL+PEC model.

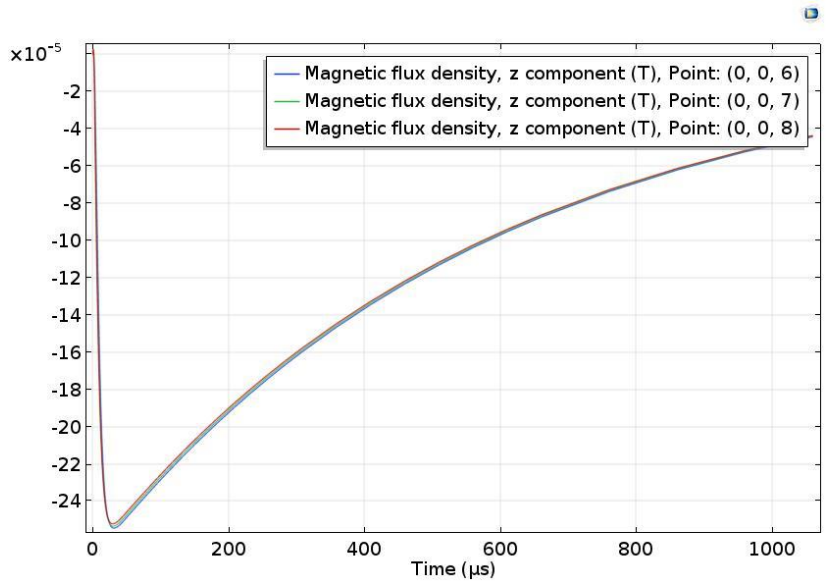


Figure 19: Magnetic flux density with lift-off 1mm,2mm and 3mm during 0 to 1000 microseconds

Modelling of PEC + MFL on comprehensive inspection

Similar to PEC setting, PEC+MFL also adopt a square wave excitation of one coil (primary coil) to induce a transient response from electromagnetic field interactions deep within the conducting structures[12]. Our present work investigates output signal of secondary coil which indicate summation of magnetic field produced by both magnets and excitation coil.

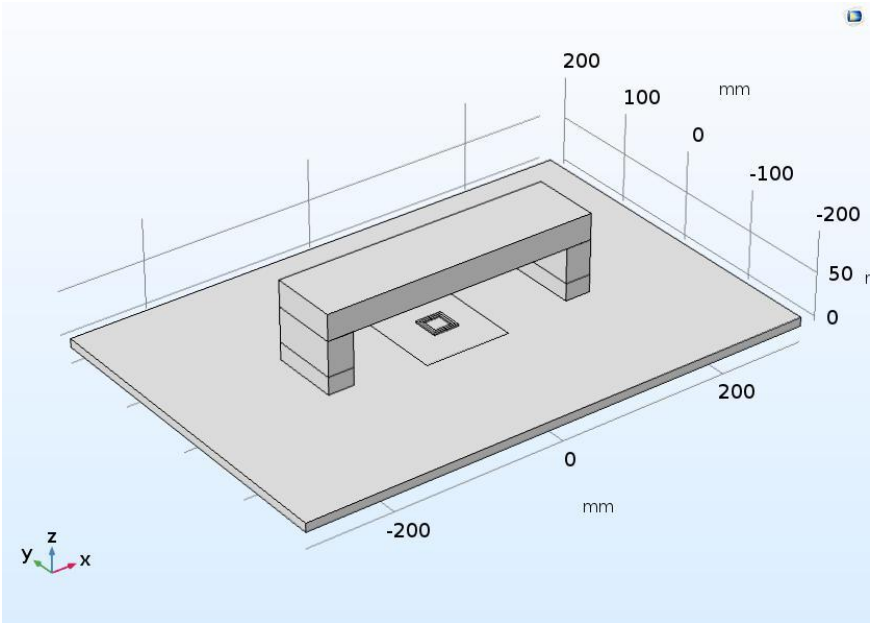


Figure 20: Schematic of PEC + MFL probe and MUT with no defect

New sensing model enable us to do faster inspection with deeper signal penetration depth as well as higher sensitivity and linearity. The core physics principle of our PEC + MFL model is detect and identify conductivity-dependent and permeability-dependent distribution pattern due to the presence of defect. However, when we compare two groups of results with first group by PEC only, second group by MFL+PEC, little difference is observed. Both groups present induced voltage value in the secondary coil and differential voltage calculated by subtracting secondary voltage from primary one.

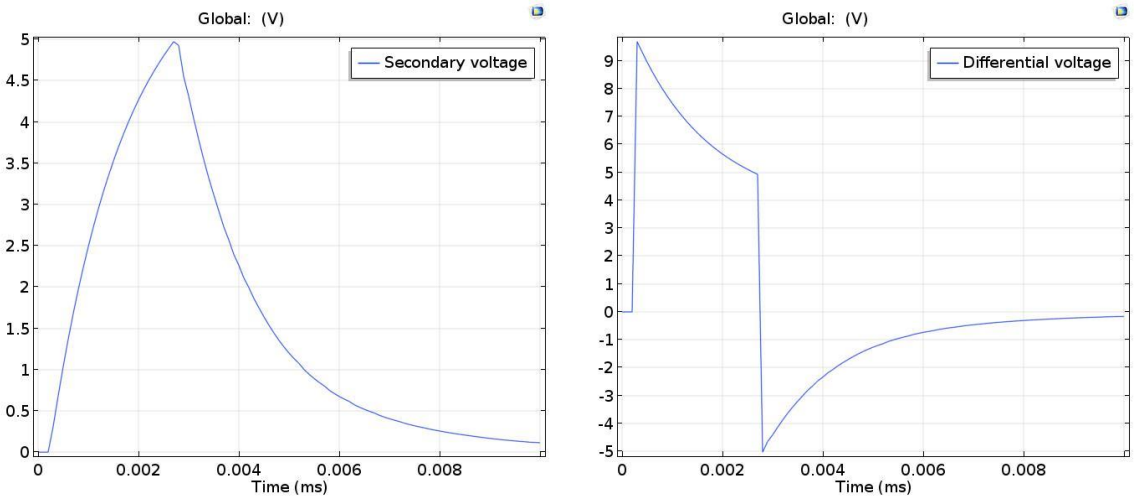


Figure 21: Induced voltage of secondary coil and differential voltage in PEC model

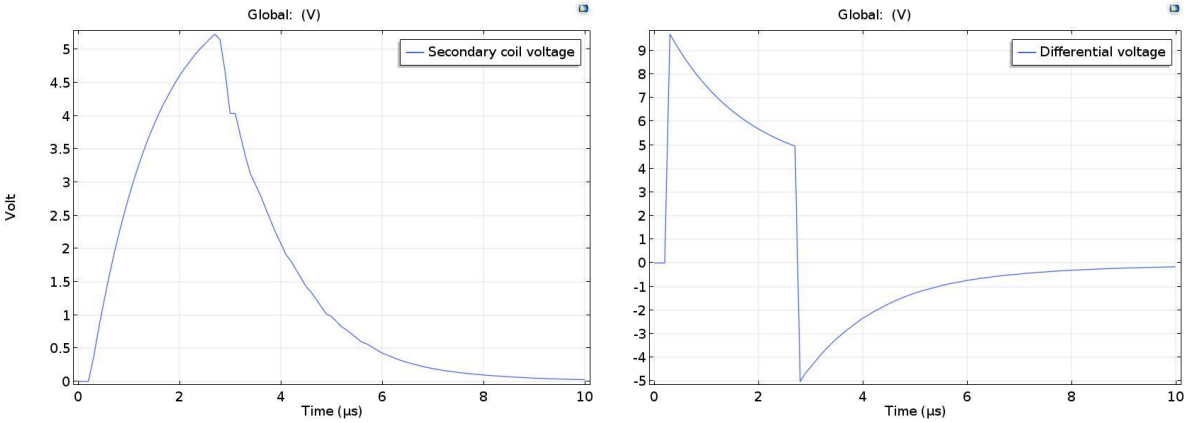


Figure 22: Induced voltage of secondary coil and differential voltage in PEC+MFL model

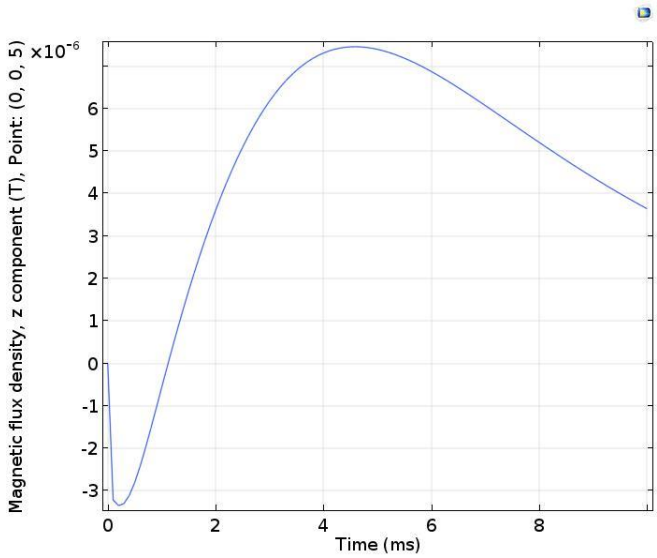


Figure 23: Magnetic flux density of sample surface during 0 to 10ms

In the experiment, we reset pulsed signal as 0 and keep MFL device functioning normally and place an imaginary sensor to measure intensity of magnetic flux on the upper surface of MUT. As illustrated in the figure, intensity value reaches its peak at 4.3ms which is much larger than time domain adopted for PEC. Thus, compatibility can be guaranteed if we get feedback from MFL and PEC separately in the time-dependent study.

2. Progress on Task 2.2: Spatiotemporal matching of interacting threats

In this subtask, we complete the spatiotemporal matching algorithm to identify the corresponding interacting thread appearing in different times. We propose to use graph matching based method to implement this goal.

For graph matching, [13] identified point correspondence by exploring principal eigenvector of the affine matrix. [14] searched correspondence through factorizing a large affine matrix into smaller matrices that encode local relationships. [15] solved the non-convex point association problem using a random walk algorithm.[16]developed a path following method to solve the optimization. Compact-ness prior was used to improve matching [17].

2.1 Approach

In our proposed method, we do not just use spatial relationship constructed by nodes in graphs, but also take the appearance feature of each node into account to identify the correspondences of nodes in two different graphs. In the following, we denote matrices by boldface, capital letters $\mathbf{B}=\{b_{ij}\} \in R^{nxm}$, we define its i-th row as b_i and vectors by lowercase letters v .

Given a pair of images of the same place but taken at different times, the information extracted from these two images are denoted as two graphs G= (V,E,C) and G'= (V',E',C'). Take graph G as an example, defect position set is denoted as $\mathbf{V}=[\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_n}] \in R^{2xn}$ which contains all the positions of detected defects, where $\boldsymbol{v_i}= [x_i, y_i]^T$, i=1,2,· · ·, n represents the central position of i-th defect in image coordinate and n is the number of defects in the image. Edge set E=

E=$[e_1, e_2, \ldots, e_m] \in R^{zxm}$, for a pairwise edge e_p= $[v_i, v_j]$, p= 1,2,· · ·, m and $v_i v_j \in$V. The attribute set **C** including all the features of defect denoted as **C**= $[c_1^T, c_2^T, \ldots, c_n^T]^T \in R^{nxd}$, where c_i^T is a feature vector obtained from the i-th defect in the image.

Based on position set **V** and edge set **E**, we can calculate the spatial relationships of the landmarks by Eq.(11), denoting as P=$\{p_{ij}\} \in R^{nxn}$, i≠j for pairwise edge and dis() function is to calculate the length of edge. Based on **C** and **C'**, we can obtain appearance similarity matrix **B**=$\{b_{ij}\} \in R^{nxn}$, where b_{ij} = cos<c_i ,c'_j >, matrix **B** and **P** are explained in Figures 24 and 25.

$$p_{ij} = exp\ (\gamma||dis(e_i) - dis(e'_j)||_2) \qquad (11)$$

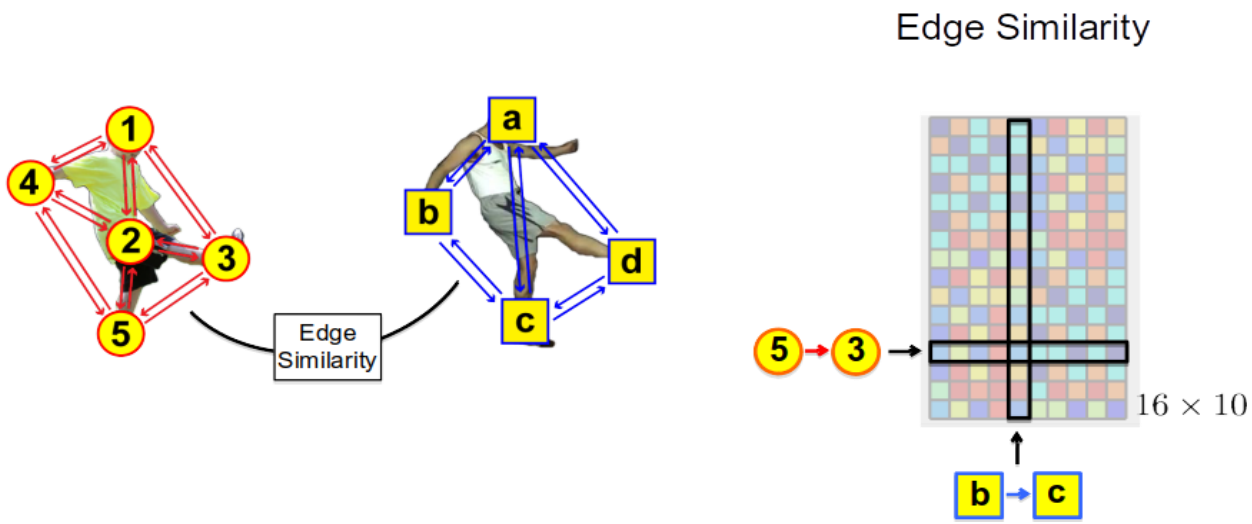


*Figure 24: Edge similarity matrix **P***

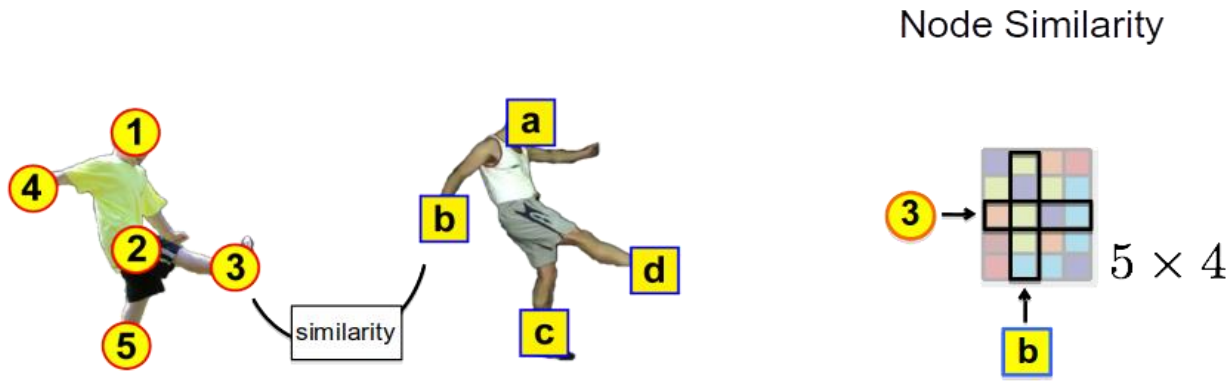


*Figure 25: Node appearance similarity matrix **B***

Finally, the matching formulation is defined as following:

$$X *= arg\ max(\sum_{ii'}^{nn'} \sum_{jj'}^{nn'} p_{ii',jj'} x_{ii'} x_{jj'} + \sum_{ii'}^{nn'} b_{ii',jj'}) \qquad (12)$$

We can rewrite Eq.(12) into matrix form as following:

$$X *= arg\ max(X^T P X\ + B^T X) \qquad (13)$$

The first term is to calculate the spatial disparity between two graphs, the second term is to calculate the similarity of defect appearances (like shape) between two graphs. The matrix **X** is the final corresponding matrix which encodes the correspondences of nodes in two graphs, as shown in the following figure.

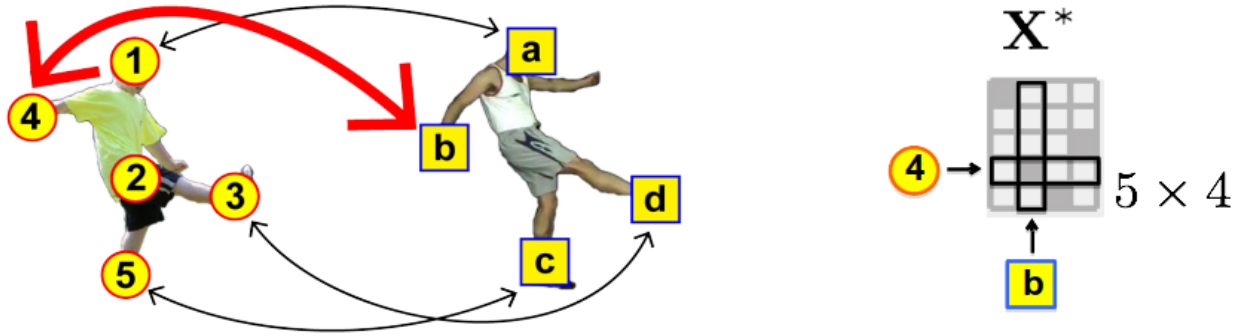


*Figure 26: Corresponding matrix **X***

2.2 Optimization

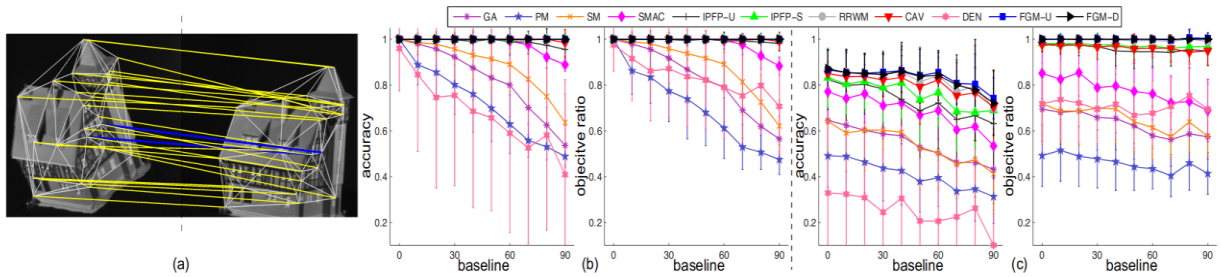


Figure 27: Comparison-1 for solver performance

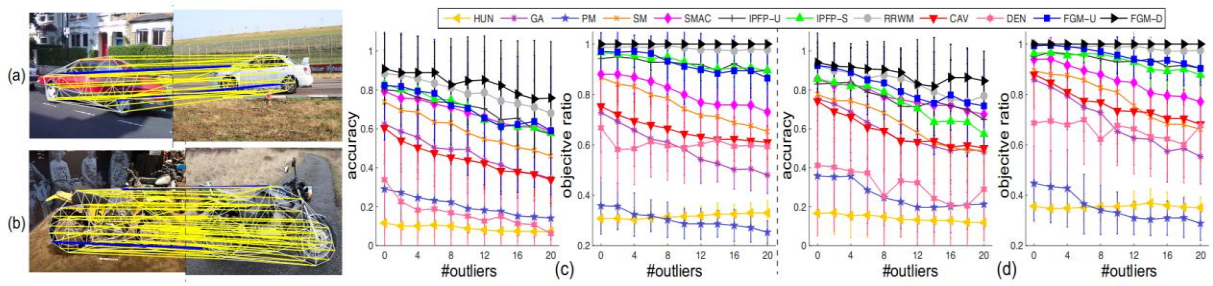


Figure 28: Comparison-2 for solver performance

Since getting the solution to formulation in Eq.(13) is NP-hard [14], there are many research try to solve it through different ways, based on the performance shown in Fig 4 [14], we develop a new solver based on RRWM [15] to solve the formulation in Eq.(13). The algorithm is as following:

Algorithm1: An algorithm to solve the formulated optimization problem in Eq.(3)

Input: P and **B**
Output: X

1. Initialize the correspondence matrix **X**
2. Computer **M** according to Eq.(14)
3. while not converge do
4. | Update **X** by Eq.(15)
5. | Computer the jump vector **Z** by Eq.(16)
6. | Normalize **Z** using bistochastic normalization
7. | Update **X** with reweighted jump by Eq.(17)
8. end
9. Discretize **X** using Hungarian Algorithm
10. Return **X**

$$m_{ii',jj'} = p_{ii',jj'} / X^T P X \quad (14)$$

In step 2 in algorithm 1, we convert the original similarity matrix to probabilistic form as shown in Eq.(14), and then update **X** by:

$$X^{r+1} = (X^T)^r M (X)^r \quad (15)$$

In order to jump out local optima, we implement a reweighting jump vector **Z** as :

$$Z = exp(X \circ B / max(X \circ B)) \quad (16)$$

where ◦ denotes the entrywise product. The node appearance similarity B is used to guide the jump toward a direction that can better matches similar objects.

Step 6 employs a bistochastic normalization to normalize each row and column in Z, thus enforcing the one-to-one correspondence. Then, in Step 7, to facilitate X to jump out of local optima, X is updated by:

$$X^{r+1} = (X^T)^r M (X)^r + (1 + \alpha) Z \quad (17)$$

where α is a hyper-parameter that controls the update rate. Since X is computed using real-valued numbers in R^{nxn}, in Step 9 after convergence, we discretize it to obtain a binary matrix $X \in \{0, 1\}^{nxn}$ using the Hungarian algorithm.

3. Summary and Future Work

For task 1.2, we have applied the mixture regression model along with transfer learning for monitoring the growth of single defects. We have implemented the data analysis on the magnetic flux densities obtained by running MFL simulations in CMOSOL as in presence of defects there will be perturbation in magnetic fields which are captured by magnetic sensors (domain probes) in simulation. At first, we have applied our mixture regression model (cluster K=2, defect and non-defect cluster) on the obtained magnetic field flux data that is not infected by noise. Points with posterior probabilities greater than 0.8 are considered in defect area. There we are successful in identifying and reconstructing the defect area. After that we have randomly increased the size of the defects (stochastically) and have added noise such that the wavelets failed to denoise them. Now on these noisy data mixture regression fails as shown earlier. Hence, we are transferring mixture regression model from noiseless data to transfer the locations and size of the defect. We classify the neighbors based on the noiseless data and thereby accurately predicted the defect growth on the noisy data. We have successfully predicted the growth of defect as shown in the above figures thereby reducing the un-certainty and increasing the accuracy. However, when the defects are interacting with each other then, to study the interaction among the defects and how they are spreading we have presented here a brief overview of kriging as there will be correlation among the threats while in interaction. In simulation part, PEC is studied first in terms of COMSOL based simulation. Excitation sources are given by external circuit which enable us to implement more complex input signals. Moreover, compatibility of PEC and MFL is verified by observation in the time-dependent study of magnetic flux leakage. In the future, we will study more comprehensive feedback signal from MFL and PEC simulations regarding spatial dependence of the threats. Postprocessing methods such as Kriging will be applied on the interaction in next quarter. As to task 2.2, we implement a pairwise graph matching method to find the correspondences between objects in different images, which consider both appearance feature and spatial relationship, so that to mimic the situation of identifying spatiotemporal defects. In order to evaluate our proposed algorithm, we collect our own dataset which used multiple different robots to represent defect. Based on our experimental result, we found that even if there exist view change or spatial deformation, our method can still find the correspondences of objects between a pair of images.

In the next quarter, we continue making progress and completing the research tasks including tasks 1.2 and 2.2, following the project schedule included in the approved proposal. We will also focus on collecting data of interacting threats and apply the developed methods on the collected dataset. To promote education, we will continue involving PhD, Master's, and undergraduate students from our research groups in the project, and advise them to improve their research skills as the project continues.

References

[1] The "Risk Modeling Work Group" Discussion of Interactive Threats. 2013.

[2] Chen, Y., et al., *Failure assessment of X80 pipeline with interacting corrosion defects.* Engineering Failure Analysis, 2015. 47: p. 67-76.

[3] America, I.N.G.A.o., *Interacting Threats to Pipeline Integrity – Defined and Explained.*

[4] Bengio, Y., *Deep Learning of Representations for Unsupervised and Transfer Learning.* JMLR: Workshop and Conference Proceedings, 2012.

[5] Tatiana Benaglia, D.C., David R. Hunter,Derek S. Young, *mixtools: An R Package for Analyzing Finite Mixture Models.* Journal of Statistical Software, 2009.

[6] Sinno Jialin Pan and Qiang Yang Fellow, I., *A Survey on Transfer Learning.* IEEE Transactions on knowledge and data engineering, 2010.

[7] Parresol, L.E.a.B., *Development of a Regression Kriging Model Conditioned with Sequential Gaussian Simulation to Predict the Spatial Distribution of Site Index for The Savannah River Site.* 2012.

[8] *Kriging and Mixed Effects Models Some Connections and a Case Study on Soil Degradation.*

[9] Bas van Stein, H.W., Wojtek Kowalczyk, Michael Emmerich, and Thomas Bäck, *Cluster-based Kriging Approximation Algorithms for Complexity Reduction.*

[10] 5.2, C., *AC/DC Module User's Guide.*

[11] Huang, S. and S. Wang, The Pulsed Eddy Current Testing, in New Technologies in Electromagnetic Non-destructive Testing. 2016. p. 41-80.

[12] Piao, G., et al., *A novel pulsed eddy current method for high-speed pipeline inline inspection.* Sensors and Actuators A: Physical, 2019. 295: p. 244-258.

[13] Leordeanu, Marius, and Martial Hebert. "A spectral technique for correspondence problems using pairwise constraints." Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1. Vol. 2. IEEE, 2005.

[14] Zhou, Feng, and Fernando De la Torre. "Factorized graph matching." 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2012.

[15] Cho, Minsu, Jungmin Lee, and Kyoung Mu Lee. "Reweighted random walks for graph matching." European conference on Computer vision. Springer, Berlin, Heidelberg, 2010.

[16] Liu, Zhi-Yong, and Hong Qiao. "GNCCP—Graduated nonconvexityand concavity procedure." IEEE transactions on pattern analysis and machine intelligence 36.6 (2013): 1258-1267.

[17] Suh, Yumin, Kamil Adamczewski, and Kyoung Mu Lee. "Subgraph matching using compactness prior for robust feature correspondence." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.